Semantic Prompting with Image Token for Continual Learning

Jisu Han[1], Jaemin Na[2*], and Wonjun Hwang[1*]
[1]Ajou University, Korea, [2]Tech. Innovation Group, KT, Korea
{jisu3709, wjhwang}@ajou.ac.kr, jaemin.na@kt.com

Abstract

Continual learning aims to refine model parameters for new tasks while retaining knowledge from previous tasks. Recently, prompt-based learning has emerged to leverage pre-trained models to be prompted to learn subsequent tasks without the reliance on the rehearsal buffer. Although this approach has demonstrated outstanding results, existing methods depend on preceding task-selection process to choose appropriate prompts. However, imperfectness in task-selection may lead to negative impacts on the performance particularly in the scenarios where the number of tasks is large or task distributions are imbalanced. To address this issue, we introduce a novel task-agnostic approach that focuses on the visual semantic information of image tokens eliminating the preceding task prediction. By leveraging the ability of the pre-trained model to discriminate between similar tokens, our method not only subdivides the prompt but also eliminates the need for additional forward pass. Consequently, we achieve competitive performance on four benchmarks while significantly reducing training time compared to state-of-the-art methods. The code is available at https://github.com/pilsHan/I-Prompt

1. Introduction

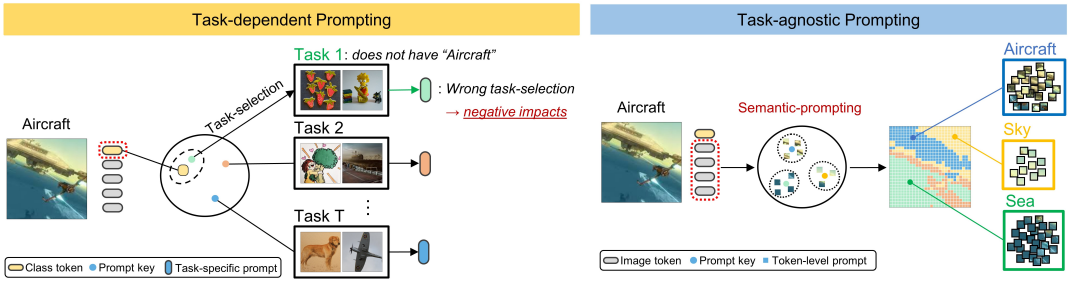
Continual learning is an adaptive approach to training deep neural networks, enabling them to adapt and evolve as they encounter new data streams over time. In contrast to traditional paradigms that generally train on static dataset, continual learning focuses on the ability of networks to continuously learn from non-stationary distributions. This approach is crucial in real-world applications where the nature of tasks can dynamically change or expand. The core challenge in continual learning is to equip the networks with the capability to integrate new knowledge while retaining previous knowledge, alleviating catastrophic forgetting [7, 24].

The rehearsal-based approach [4, 32, 44], widely used in continual learning, aims to mitigate the loss of prior knowledge by periodically retraining the neural network on a subset of previous data using a memory buffer. Although the rehearsal-based methods have demonstrated impressive results in addressing catastrophic forgetting problem, they also raise concerns regarding data privacy and the requirement for additional memory buffers. Consequently, there is a growing demand for the development of a more efficient, rehearsal-free approach [28, 50–52] that can achieve comparable performance than current rehearsal-based methods.

Recently, prompt-based methods [35, 41–43] have shown competitive performance and superior efficiency compared to rehearsal-based methods, utilizing only a few model parameters without any memory buffers. These methods enhance the performance by training the carefully designed prompts, while keeping the remaining parameters frozen. In particular, L2P [43] constructs a prompt pool and match the output of a pre-trained model as a query to select appropriate prompts from the pool. Building upon this foundation, recent works [35, 42] improve efficacy by optimizing the positions of the prompts within the model and refining key-query mechanism. These methods have achieved remarkable success; however, these methods focus on the task and design task-specific prompts, which involves a task prediction process to select the trained prompts for each task. Therefore, in contrast to the learning process, the selection of prompts that are not trained for the input class due to incorrect task prediction in the inference process where the task ID is unknown leads to forgetting. In soft-selection using weighted combination, forgetting is also caused by the inconsistency of the training and inference process. In particular, as the task prediction becomes more difficult, such as the number of classes per task is imbalanced or the boundary between tasks is blurred, the performance decreases due to the wrong prompt selection and it is difficult to be adaptable to various scenarios.

In this work, we introduce I-Prompt, an image token-based semantic prompting method that exploits the inherent semantic information of image classes. As depicted in Figure 1, we mitigate the risk of selecting the wrong task by eliminating the traditional task selection process. Instead,

* Corresponding authors.



(a) Previous approach (b) I-Prompt (Ours)

Figure 1. **Overview of the prompt-based continual learning approaches.** (a) Previous approaches [41, 42] have selected prompts based on the output class token of the pre-trained model and their similarity to task-specific prompts, accompanied by a task selection process. If an aircraft image is properly allocated to task T, which involves aircraft, accurate inference can be expected. However, if it is assigned to task 1, it leads to forgetting due to the inconsistency between training and inference. (b) Our approach eliminates these erroneous task-selection process and focuses on semantic information within image itself to assign prompts that are relevant to the image. We exploit the information of relationships between image-tokens through the representational capability of the pre-trained model.

we prioritize the use of image tokens to effectively harness the rich semantic information contained within the image itself. Our work stems from empirical studies on vision transformer [5], which have demonstrated the effectiveness of exploiting the attention structure in transformer layers for clustering similar tokens [23]. Similarly, we are inspired by previous work [2] that had showed token similarity can be efficiently computed using only self-attention key, which reduces computational costs. Building on these insights, we introduce a prompting method that not only utilizes token similarities within images, but also leverages information about class-specific visual characteristics, offering a more efficient approach.

In continual learning, the prompt query is designed to be applied differently depending on the characteristics of the data to overcome forgetting. In traditional methods, the role of class tokens is to predict the task to which they belong, leveraging the zero-shot classification ability of pre-trained models. Our motivation lies in employing the classification ability of the tokens in the attention structure. This allows us to exploit visual features in the intermediate process, rather than predicting the task by replacing the class token. Moreover, by replacing the task prediction process, we not only reduce the task dependency, but also reduce the additional training cost of the prompt selection process. The traditional method of using the class token as a query for the prompt requires an additional forward pass in the prompt selection process to use the final output of the model, resulting in the inefficiency of performing two forward passes. In contrast, our method simplifies this process by selecting the prompts within the transform layers effectively eliminating the need for extra forward pass. Consequently, our approach enables training and inference only with a single forward pass since the prompt selection and prediction process is conducted simultaneously.

We conducted extensive ablation studies for a detailed analysis of the proposed methods. In particular, we achieve competitive performance to the recent state-of-the-art methods in standard task-balanced benchmarks such as CIFAR-100 [17], CUB-200 [39], and ImageNet-R/A [10, 11]. Furthermore, we demonstrate the effectiveness of our task-agnostic approach in task-imbalanced scenarios, particularly on the ImageNet-R and CIFAR-100 benchmarks.

2. Related work

2.1. Continual Learning

Continual learning aims to enhance performance by acquiring new tasks while minimizing the forgetting of knowledge from previous tasks. Representative methods for solving the forgetting problem in continual learning include regularization-based methods, rehearsal-based methods, and dynamic architecture methods. Regularization-based methods [1, 3, 15, 45] determine the importance of model parameters for previous tasks and then apply strong regularization to these important parameters while using less important parameters for learning new tasks. It mitigates forgetting by making the change in loss to the previous task small while learning new tasks with less important parameters. This approach offers the benefit of reduced memory requirements for continuous tasks; however, it faces the challenge of diminished performance, attributed to updating model parameters without direct access to the data for previous tasks. Meanwhile, rehearsal-based methods [20, 30, 32, 44] have achieved high performance by mitigating the forgetting problem through the limited size of memory buffers for the previous tasks. However, they pose additional memory requirements and raise privacy and security concerns due to the storage of past task data. On the other hand, dynamic architecture methods [22, 34, 40] freeze

models from previous tasks and add sub-networks for the new tasks. This method effectively avoids the forgetting problem but results in a linear increase in model parameters with each new task, potentially leading to less manageable models in scenarios with various tasks.

2.2. Prompt-based Continual Learning

A method for adapting to downstream tasks without updating the model has been proposed in the field of Natural Language Processing (NLP), focusing on finetuning the large language models [18, 19] using learnable prompts. This success in NLP tasks is extended to vision tasks that require parameter-efficient finetuning in recent studies [12, 36, 38]. In continual learning, prompt-based methods [35, 42, 43] are included in the dynamic architecture method in that there is an additional parameter called a prompt in addition to the model parameters. L2P [43], the first study to employ prompts in continual learning, achieves meaningful results by selecting prompts through query-key matching in the query function and learning only the prompts, using them as additional inputs to the model. DualPrompt [42] introduces a task-invariant and task-specific prompts for complementary learning. S-Prompt [41] proposes task-specific prompts for domain incremental learning, and predicts domains via k-means clustering. Furthermore, CODA-Prompt [35], points out the limitation in query-key matching, where the gradient does not flow end-to-end, and addresses this by enhancing learnability through end-to-end training of a prompt directly from the classification loss. In parallel to prompt-based methods, Adaptor-based methods [25, 29, 47, 48] are also investigated in the context of subspace as a method for efficient continual learning. Furthermore, language-guided prompt-based approaches [13, 49] have been studied, but they require an additional memory usage for text encoder.

2.3. Token Similarity in Transformer

Research on efficient transformers [6, 9, 26] is underway to reduce redundant calculations and enable faster calculations by downsampling using similarities between tokens. Token pooling [23] shows that the attention layer of the vision transformer generates overlapping tokens and proposes a method for selecting and pooling similar tokens via clustering from features. Furthermore, Token merging [2] focuses on the self-attention mechanism of the transformer and explains that the key, calculated by cosine similarity for the query, inherently contains token information. Our method is motivated by findings in the literature on efficient transformers that cluster and token similarity can be calculated through the attention structure in a transformer. We improve efficiency by obtaining the query on which the prompts are selected from inside the transformer layer, instead of using output of a pre-trained transformer encoder.

3. Method

3.1. Preliminary

Continual learning protocol. Continual Learning sets up a learning scenario for sequential tasks $\mathcal{T} = \{1, 2, 3, ..., T\}$, where T denotes the number of total tasks. Model consists of a feature extractor and a classifier, each of which is a parametric model with θ and ϕ as parameters. Model prediction $\hat{y} = f_\phi \cdot f_\theta(x)$, where f_θ and f_ϕ are the feature extractor and classifier, respectively. We aim to achieve high performance on the test data $D^{1:t} = \{x^{1:t}, y^{1:t}\}$ of all previously trained tasks while training on the current task data $D^t = \{x^t, y^t\}$ for sequential tasks, where x and y denote image and label, respectively. Following existing continual learning studies [35, 42, 43], we focus on class incremental learning scenario. In the class incremental learning scenario, it is assumed that classes for different tasks do not overlap ($y^t \cap y^{1:t-1} = \varnothing$), and there is no information about which task is in the test process.

Vision transformer. Vision Transformer (ViT) tokenizes the input image, divides it into patches, and then goes through the embedding layer and positional encoding as input to the transformer layer. Therefore, where the feature for l-th layer is h_l, the initial input to the transformer layer is expressed as $h_0 = [\mathrm{CLS}; \mathrm{IMG}_1, \mathrm{IMG}_2, ..., \mathrm{IMG}_\mathrm{p}]$, where p is number of patches. The inside of the transformer layer consists of a Multi-Layer Perceptron (MLP), LayerNorm (LN), Multi-Head Self-Attention (MHSA) and residual connection. The process for each layer is conducted as follows:

$$\begin{aligned} &h_{l+1} = \mathrm{MLP}(\mathrm{LN}(z_{l+1}) + z_{l+1}), \\ &where\ z_{l+1} = \mathrm{MHSA}(\mathrm{LN}(h_l)) + h_l. \end{aligned} \tag{1}$$

The overall ViT process consists of three steps. First, the input image goes through tokenization and positional encoding for location information. After the class token is combined with the image token obtained, the encoder output is determined through several transformer layers. Finally, the class token from the encoder output is used as input of the classifier to compute the final prediction.

Traditional prompt matching. Prompt-based continual learning uses pre-trained ViT as the feature extractor. Since pre-trained models produce consistent output for the same input, existing studies use this process as a query function to select a task-wise prompt. Query-key matching method [35, 43] selects the prompt with the maximum similarity between the query and the prompt key obtained by query function.

$$K_s = \underset{i \in \tau}{\mathrm{argmax}}\ \gamma(q(x), k_i), \tag{2}$$

where K_s is selected task-specific prompt key and is an element of the prompt key set $K = \{k_1, k_2, ..., k_T\}$, $\tau =$

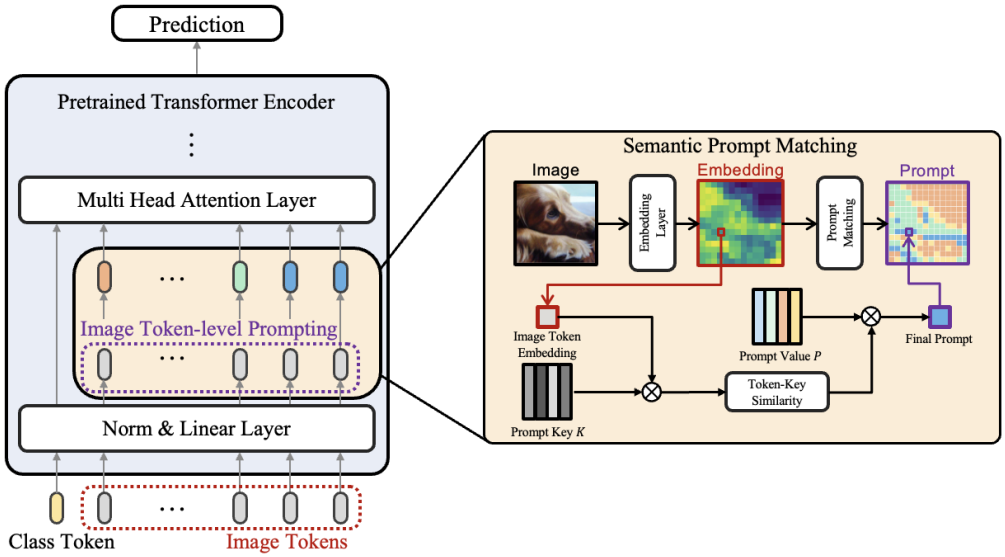


Figure 2. **Schematic illustration of I-Prompt. (Left):** Within the internal process of the transformer layer, prompts are determined at the image token level. The determined prompts are then added to the image tokens, becoming the subsequent input. **(Right):** The process of matching prompts in the prompt pool. The similarity between the input attention key from the transformer layer and the prompt key is calculated, and the final prompt is determined by the element-wise product of the calculated similarity and the prompt.

$\{1, 2, 3, ..., t\}$ is a subset of $\mathcal{T}$ and is the set of tasks up to the current task t. Query function $q(x) = f_\theta(x)[\text{CLS}]$ is the class token of the ViT encoder output, and $\gamma(\cdot, \cdot)$ denotes cosine similarity. On the other hand, attention-based method [35] performs soft selection using attention rather than hard selection through similarity between queries and keys, and match prompts by their weighted combination. Then, the attention-based prompt $\mathbf{P}_a$ is defined as follows:

$$\mathbf{P}_a = \sum_{i \in \tau} \gamma(q(x) \odot A_i, k_i)\, P_i, \tag{3}$$

where prompt P_i is element of prompt pool $P = \{P_1, P_2, ..., P_T\}$, A_i is an attention, which represents a learnable parameter for determining the weighted combination of prompts, and $\odot$ denotes element-wise product.

3.2. Semantic Prompting with Image-token

We aim to develop a task-agnostic method for prompt-based continual learning that eliminates the task prediction process. To achieve this, we select the prompts by focusing on image tokens in the internal structure of the transformer layer, rather than class token of the query function.

Semantic prompt matching. The self-attention structure in Transformer replicates the query-key-value through a linear layer, and attention to the value is determined based on the cosine similarity between the query and key. In this process, the key in the self-attention can be used as a judge that containing semantic information about the token, which we use as a prompt query for the input token. In order to ensure that similar prompts are assigned to visually similar tokens, the similarity of the prompt keys is calculated based on the self-attention key and used as a weight for each prompt. The calculated weight of the prompt is multiplied by the prompt, and the final prompt for each image token is determined by their summation. Our final prompt $\mathbf{P}$ is defined as follows:

$$\mathbf{P} = \sum_{i \in \tau} \gamma(h_k, k_i) \odot P_i, \tag{4}$$

where prompt key k_i and prompt P_i are learnable parameters, the self-attention key $h_k = W_k h$ and W_k is weight of self-attention key. The final prompt $\mathbf{P}$ is split into two independent prompt $\{P_k, P_v\} \in \mathcal{R}^{\frac{L_p}{2} \times d}$, where L_p and d denote prompt length and feature dimension, respectively, which are computed with the key and value of the self attention Semantic prompt matching on similarity between tokens selects prompts through visual representation and thus achieves task-agnostic prompting by focusing on class classification rather than task prediction.

We counteract catastrophic forgetting by fixing previously learned prompts, learning new prompts, and then boosting them. In continual learning, fixing parameters relative to previous learning prompts is the simplest and effective method to deal with catastrophic forgetting [31, 35, 41]. However, taking the prompts for all tasks as input leads to a linearly increasing number of input prompts depending on the task. Additionally, considering only the current task fails to take into account class relationships due to the isolation between tasks. Hence, we propose a more robust

prompt $\bar{\mathbf{P}}$ inspired by the boosting algorithm [40].

$$\bar{\mathbf{P}} = \gamma(h_k, k_t) \odot P_t + \sum_{i=1}^{t-1} \gamma(h_k, \tilde{k}_i) \odot \tilde{P}_i. \quad (5)$$

Note that $\tilde{P}_i$ and $\tilde{k}_i$ are fixed parameters and P_t and k_t are learnable parameters. Our method achieves a balance between stability and plasticity by carefully learning prompts by fixing the parameters for the previous task and merging the residual for the newly learned task.

Image token-level prompting. Prompt tuning methods are divided into prompt-tuning [18] and prefix-tuning [19] for input depending on the application location of the prompt.

$$\text{Prompt} : h_0 = [\text{CLS}; \text{IMG}_1, \text{IMG}_2, ..., \text{IMG}_\text{p}; \mathbf{P_s}], \quad (6)$$

$$\text{Prefix} : z_{l+1} = \text{MHSA}(\text{LN}([h_l; \mathbf{P_s}])) + h_l, \quad (7)$$

where $\mathbf{P_s}$ is the selected prompt, and is determined from the prompt pool. Previous continual learning methods for assigning batch or instance-level prompts are based on these two methods. In contrast, in our method of applying token-level prompts, concatenating prompts on input is inefficient. For computational efficiency and to take advantage of the prompts selected at the image token level, we adopt a method that directly adds the prompts to the image token:

$$\text{I-Prompt} : z_{l+1} = \text{MHSA}(\text{LN}([h_l \oplus \bar{\mathbf{P}}])) + h_l, \quad (8)$$

where $\oplus$ denotes element-wise sum. We apply our I-Prompts to the image tokens through a relatively lightweight sum operation, without any dimensional expansion for multiplicative operations. This allows for efficient computation despite our allocation of token-level prompts.

Objective function. We optimize the classifier weight ϕ, prompt P and prompt key K pairs for the fixed model parameters $\tilde{\theta}$. Finally, our objective function is as follows:

$$\underset{P,K,\phi}{\text{argmin}} \ \mathcal{L}_{cls}(f_\phi(M \cdot \hat{h}), \ y), \quad (9)$$

where $\mathcal{L}_{cls}$ is cross-entropy loss for classification and M is a logit mask, which replaces the logits for classes not included in the training data with negative infinity. Note that the logit mask [35,42,43] helps prevent forgetting about previously learned classes by stopping the gradient flow of the classifier for classes that are not training.

4. Experiments

Datasets. We evaluate our method on continual learning benchmarks such as CIFAR-100 [17], CUB-200 [39], renditions of ImageNet [33], specifically ImageNet-R/A [10,11].

- **CIFAR-100** is a widely used benchmark for continual learning and contains 100 classes. The train set and test set are split into 50,000 and 10,000 images respectively and consist of the same number of data for all classes.
- **CUB-200** is organized into subcategories of birds and contains 200 classes. The training set consists of 9,430 images, and the test set consists of 2,358 images.
- **ImageNet-R** includes images from various domains to be closer to the real problem. It contains 24,000 training images and 6,000 images for test set. It contains 200 classes.
- **ImageNet-A** consists of naturally occurring examples that are incorrectly predicted by models pre-trained with ImageNet and includes 200 classes. The train set and test set are split into 5,981 and 1,519 images.

Evaluation scenarios. As a continual learning scenario, we conduct experiments on task-balanced and task-imbalanced scenarios. Task-balanced scenarios are traditional evaluation protocols where the class is equally divided between each task, while task-imbalanced scenarios are settings where the number of classes per task is not constant, including a scenario where half of the class learns on the initial task, and a random increase scenario where the incremental class changes dynamically. When the initial learning class is *X* and the increasing number of classes is *Y*, we denote by *BX-IncY*. For example, an experiment that trains 50 classes as base and increases by 10 classes is denoted as *B50-Inc10*.

Evaluation metrics. We report Avg-Acc and Last-Acc as evaluation metrics. Let A_t be the average accuracy for all classes in task t. where Avg-Acc is the average of the accuracy on each task ($\frac{1}{T}\sum_{t=1}^{T} A_t$), and Last-Acc is the accuracy for all classes on the final task (A_T).

Implementation details. Our whole experiments are based on PILOT [37] and conducted with NVIDIA RTX A6000. We use the Adam [14] optimizer and adjust learning rate using a cosine scheduling [21]. In accordance with the evaluation protocol outlined in CODA-Prompt [35], we utilize the ViT-B/16 model pretrained on ImageNet-1K, with a shuffled class order. Additionally, due to the absence of results for various scenarios from existing methods, we provide reproduction results based on their official implementations.

4.1. Comparison with State-of-the-Arts

In our experiments, we compare with rehearsal-free prompt-based methods [35, 41–43]. We also present rehearsal-based methods [32,40,44] and two baselines: upper bound and lower bound. Joint-Training serves as the upper bound of accuracy, representing the method of learning all classes at once, while Finetuning serves as the lower bound of accuracy, representing the method of learning new tasks without any regularization.

Task-imbalanced scenario. Table 1 shows the results for the task-imbalanced scenario where the distribution of classes per task is uneven. In this setup, half of all classes are initially trained, and then the remaining classes are split according to tasks. Overall, our method demonstrates superior performance on both the ImageNet-R and CIFAR-100

Table 1. **Comparison results (%) in task-imbalanced scenarios.** The methods under comparison are divided into two categories: rehearsal-based and rehearsal-free approaches, with the best accuracy highlighted in bold.

Method	Exemplar size	ImageNet-R						CIFAR-100					
		B100-Inc5		B100-Inc10		B100-Inc20		B50-Inc2		B50-Inc5		B50-Inc10	
		Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
Joint-Training	-	-	81.58	-	81.58	-	81.58	-	92.33	-	92.33	-	92.33
iCaRL	20/class	68.48	60.35	68.60	60.33	71.90	64.62	80.46	68.87	83.85	73.63	86.53	79.71
BiC	20/class	73.20	68.92	75.41	71.93	76.84	74.18	81.06	73.96	86.21	80.84	88.41	84.74
Foster	20/class	80.45	77.17	80.13	76.55	79.88	76.60	90.83	87.88	90.67	88.56	90.47	87.80
Finetuning	0/class	59.63	47.37	64.08	57.07	72.49	61.67	67.86	60.21	78.61	69.06	81.14	73.39
L2P	0/class	64.07	52.65	68.84	59.58	73.16	66.63	67.67	49.80	80.21	69.27	86.78	80.57
DualPrompt	0/class	62.36	54.03	66.47	59.82	70.15	65.03	68.81	52.98	81.79	73.44	86.03	80.84
S-Prompt	0/class	70.56	63.97	73.88	69.32	76.13	72.70	71.38	56.10	83.66	75.87	87.82	82.52
CODA-Prompt	0/class	71.24	64.20	75.75	70.88	77.88	73.92	74.58	61.23	85.21	78.23	90.07	85.26
I-Prompt (Ours)	0/class	**73.96**	**67.30**	**78.01**	**73.30**	**79.23**	**75.82**	**75.10**	**63.26**	**86.66**	**80.75**	**90.32**	**87.09**

Table 2. **Comparison results (%) in task-balanced scenarios.** The comparison methods are categorized into rehearsal-based and rehearsal-free methods.

Method	Exemplar size	ImageNet-R						CIFAR-100					
		B0-Inc10		B0-Inc20		B0-Inc40		B0-Inc5		B0-Inc10		B0-Inc20	
		Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
Joint-Training	-	-	81.58	-	81.58	-	81.58	-	92.33	-	92.33	-	92.33
iCaRL	20/class	68.50	56.28	71.82	61.43	75.65	65.60	82.71	69.52	85.30	74.60	87.61	79.20
BiC	20/class	78.49	71.57	80.21	74.85	81.05	76.17	85.21	77.30	88.52	82.72	90.71	86.37
Foster	20/class	83.00	76.52	82.46	76.27	82.32	75.73	91.87	87.22	92.08	87.83	91.41	86.80
Finetuning	0/class	66.75	48.23	71.35	61.40	76.24	64.78	75.83	63.86	79.50	68.01	84.96	75.65
L2P	0/class	75.91	70.13	78.11	72.63	78.78	74.62	84.18	77.70	89.26	84.41	90.54	85.85
DualPrompt	0/class	72.52	66.00	74.94	69.13	74.51	70.05	84.88	77.39	87.39	82.38	88.10	83.46
S-Prompt	0/class	72.90	65.98	75.67	69.72	77.72	72.22	82.34	72.25	87.45	80.64	90.20	85.41
CODA-Prompt	0/class	78.65	72.18	81.44	75.08	81.46	76.72	88.03	80.66	91.45	86.19	92.52	88.40
I-Prompt (Ours)	0/class	**79.74**	**73.22**	**81.75**	**75.73**	**81.86**	**76.92**	**89.69**	**84.62**	**91.75**	**87.63**	**92.69**	**88.91**

Table 3. **Comparison results (%) in task-balanced scenarios.** Comparison methods are divided into rehearsal methods and rehearsal-free methods.

Method	Exemplar size	ImageNet-A						CUB-200					
		B0-Inc10		B0-Inc20		B0-Inc40		B0-Inc10		B0-Inc20		B0-Inc40	
		Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
Joint-Training	-	-	56.81	-	56.81	-	56.81	-	87.79	-	87.79	-	87.79
iCaRL	20/class	49.12	37.33	40.15	32.13	49.09	39.83	89.59	82.44	89.81	84.56	89.07	83.50
BiC	20/class	50.12	38.25	47.21	38.31	50.55	40.75	88.14	83.59	89.19	83.72	89.91	85.33
Foster	20/class	60.28	50.30	57.55	50.63	55.97	49.64	85.26	82.70	81.48	77.65	78.28	70.53
Finetuning	0/class	30.91	13.63	36.94	20.80	44.91	26.27	57.66	36.34	69.80	52.12	77.52	65.31
L2P	0/class	48.73	38.78	53.10	44.70	54.82	48.12	70.02	58.06	77.27	66.54	79.90	71.46
DualPrompt	0/class	55.20	43.05	57.81	47.07	59.77	50.76	**77.52**	64.80	79.91	69.17	81.57	72.73
S-Prompt	0/class	48.85	35.88	56.51	44.50	60.47	50.10	76.28	63.15	81.57	70.57	85.57	77.78
CODA-Prompt	0/class	54.33	44.63	62.85	52.40	65.74	56.22	76.77	**66.58**	83.39	73.20	85.76	77.78
I-Prompt (Ours)	0/class	**62.28**	**50.76**	**65.71**	**55.83**	**66.48**	**56.48**	77.29	66.07	**84.25**	**74.64**	**85.81**	**78.67**

benchmarks, outperforming other state-of-the-art methods. Specifically, in ImageNet-R, we achieve a significant performance improvement of 2.72% in average accuracy and 3.30% in last accuracy compared to the existing best accuracy in the B100-Inc5 setting. In CIFAR-100, we obtain performance enhancements of up to 1.45% and 2.52% for average and last accuracy, respectively, in the B50-Inc5 setting. Performance increases significantly as the total number of tasks increases. This tendency satisfies our objective of mitigating the forgetting problem that occurs as task prediction becomes more difficult.

Task-balanced scenario. We show the experimental results for the task-balanced scenario in Tables 2 and 3. The task-balancing scenario splits all tasks into an equal number of classes. The task-balanced scenario is the basic experimental setting of the previous work [35, 42, 43], and there is no task imbalance problem. Our method achieves competitive performance on CIFAR-100, ImageNet-R/A, and CUB-200 compared to prompt-based methods. Especially, for CIFAR-100 B0-Inc5, it achieves 1.66% and 3.96% higher performance in terms of average accuracy and last accuracy than before, and in ImageNet-A, which has the largest per-

Table 4. **Results (%) on online continual learning scenario.** A higher AUC-Acc indicates that model's ability to consistently maintain high accuracy while adapting to new task over the training. † denotes our reproduced results with their official codes.

Method	CIFAR-100		ImageNet-R	
	AUC-Acc	Last-Acc	AUC-Acc	Last-Acc
Finetuning	19.71 ± 3.39	10.42 ± 4.92	7.51 ± 3.94	2.29 ± 0.85
Linear Probing	49.69 ± 6.09	23.07 ± 7.33	29.24 ± 1.26	16.87 ± 3.14
L2P	57.08 ± 4.43	41.63 ± 12.73	29.65 ± 1.63	19.55 ± 4.78
DualPrompt	67.07 ± 4.16	56.82 ± 3.49	40.11 ± 1.27	29.24 ± 4.63
MVP	$\mathbf{68.10 \pm 4.91}$	62.59 ± 2.38	40.60 ± 1.21	31.96 ± 3.07
MVP†	67.13 ± 5.05	63.10 ± 1.61	38.39 ± 1.54	31.01 ± 3.72
I-Prompt (Ours)	67.23 ± 5.76	$\mathbf{63.42 \pm 1.48}$	$\mathbf{41.08 \pm 1.54}$	$\mathbf{33.27 \pm 2.86}$

Table 5. **Results (%) with transfer-based method.** We present the number of tuning parameters and performance of the transfer-based method with I-Prompt.

Method	Tuning Parameters	CIFAR-100 B0-Inc10		ImageNet-R B0-Inc20	
		Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
I-Prompt	1.23M	91.75	87.63	81.75	75.73
SimpleCIL	0	82.31	76.21	67.06	61.28
SimpleCIL + I-Prompt	1.23M	**91.25**	**86.69**	**79.61**	**72.48**
FeCAM	0	82.65	76.64	62.92	57.15
FeCAM + I-Prompt	1.23M	**89.91**	**85.63**	**69.25**	**63.75**
RanPAC	2.09M	**94.41**	90.95	83.28	78.15
RanPAC + I-Prompt	2.02M	94.40	**91.15**	**83.43**	**78.28**
SLCA	85.9M	93.96	89.98	84.71	78.50
SLCA + I-Prompt	87.0M	**94.30**	**90.68**	**84.87**	**80.55**

formance gap, it achieves a high average accuracy of up to 7.08% difference. Moreover, it achieves notable performance even when compared to rehearsal-based methods, and even achieves better performance on CIFAR-100 B0-Inc20. In the task-balanced scenario, the performance improvement is large compared to the comparison method in an experimental environment with many tasks, which is the same as the tendency of the imbalanced scenario.

Online continual learning scenario. In Table 4, we validate our method in Si-Blurry [27] scenario characterized by stochastic blurry task boundary, which introduces a more challenging online continual learning scenario. For this experiment, we adopt AUC-Acc [16] as a metric to evaluate the efficacy of the methods. As a result, we demonstrate the robustness of our approach, consistently achieving strong AUC-Acc and Last-Acc results for both CIFAR-100 and ImageNet-R. We obtain the best performance in both metrics for ImageNet-R and the best Last-Acc with the second-best AUC-Acc for CIFAR-100. Remarkably, our method achieves competitive AUC-Acc performance without explicitly addressing the inherent class imbalance problem in this scenario. This demonstrates the applicability of our method in real-world scenarios.

4.2. Further analysis

Comparison with transfer-based Methods. Transfer-based methods achieve high performance in continual learning by the simple and effective refinement of the classifier through generalized embedding of a pre-trained model. SimpleCIL [48] constructs a classifier without training by using class mean vectors, while FeCAM [8] applies classification based on the moments of each class vector. RanPAC [25] trains the adapter [26] on the first task and introduces a Random Projection layer with additional dimensions between the embedding and classifier. SLCA [46] aligns the classifier using mean and covariance, while updating the model parameters with a small learning rate during training. Furthermore, prompt-based methods that adjust embedding can be plug-and-played without conflict with transfer-based methods that refine classifier. Consequently, we observed consistent performance improvements when integrating our method with SimpleCIL, FeCAM, and SLCA, and achieving comparable performance on RanPAC.

Table 6. **Task prediction analysis.** We report task prediction accuracy and last accuracy on ImageNet-R B0-Inc20.

Method	Task-Acc	Last-Acc
DualPrompt-Query	55.8	68.13
DualPrompt-Perfect Match	100	71.97
S-Prompt-Query	42.6	69.72
S-Prompt-NCM	57.7	63.43
S-Prompt-Perfect Match	100	85.12
I-Prompt	No task prediction	75.73

Discussion on task prediction process. We demonstrate the impact of task prediction accuracy on final performance in the Table 6. The Task-acc denotes the task prediction accuracy, and we observe that the performance decreases as the task is incorrectly selected by the query key mechanism compared to predicting the task correctly as an oracle. DualPrompt [42] utilizes both task-invariant and task-specific prompts, resulting in relatively small performance differences even when task predictions are incorrect. However, its overall performance is lower when the task is correctly predicted, and it remains vulnerable to catastrophic forgetting. On the other hand, S-Prompt [41], which relies solely on task-specific prompts, is more resistant to forgetting when the task is correctly predicted. However, as the accuracy of task prediction decreases, performance drops dramatically. These findings experimentally confirm that methods relying on task prediction suffer from compounded errors in both task prediction and subsequent classification, leading to reduced overall performance.

Efficiency comparison. Table 7 shows the number of training parameters compared to total model parameters as additional memory usage and the learning and inference time as computation cost. Compared to the previous best performance, our method achieves the best accuracy with less than half the number of trainable parameters. For training time and inference time, we achieve a 40-50% reduction compared to CODA-Prompt, which also achieves the best performance in comparison to L2P and DualPrompt. Our method requires only one forward pass in the training and

Table 7. **Efficiency comparison.** We provide the number of training parameters, training and inference times, and accuracy.

Method	Tuning Parameters↓ (learnable/total)	Training Time↓ (ms/image)	Inference Time↓ (ms/image)	Last-Acc↑ (%)
Finetune	100%	6.58	0.071	68.01
L2P	**0.14%**	8.15	0.130	84.41
DualPrompt	0.39%	7.44	0.135	82.38
CODA-Prompt	4.57%	9.47	0.149	86.19
I-Prompt (Ours)	1.43%	**5.86**	**0.088**	**87.63**

Table 8. **Stability-plasticity analysis.** We present the results of Forgetting and Intransigence measures.

Method	Forgetting Measure ↓ (%)			Intransigence Measure ↓ (%)		
	B0-Inc5	B0-Inc10	B0-Inc20	B0-Inc5	B0-Inc10	B0-Inc20
L2P	5.61	4.80	7.91	13.64	8.97	7.47
DualPrompt	9.04	6.12	6.35	14.06	9.49	7.87
CODA-Prompt	5.08	4.05	5.12	10.84	6.82	**4.22**
I-Prompt (Ours)	**4.40**	**4.01**	**4.84**	**9.04**	**6.27**	4.32

inference process, while previous methods require an additional forward pass to obtain the query as a selection criterion for the prompt, for a total of two forward passes. This allows our method to train and inference efficiently.

Stability and plasticity analysis. Achieving a balance between stability and plasticity is an important in continual learning. To analyze the proposed method in terms of stability and plasticity, we show the forgetting and intransigence measure in Table 8. The forgetting and intransigence measures [3] are an estimate the amount of the model has forgotten and refers to the degree to which a model cannot learn a new task, respectively. Lower is better for both measures, with a low forgetting measure means high stability and low intransigence measure means high plasticity. We measure the stability and plasticity of our method on CIFAR-100 and achieve performance that outperforms existing methods. We confirm that the proposed method leads to high performance on plasticity because it directly changes the input tokens instead of changing them indirectly by concatenating additional inputs to the prompt, and achieves high stability by learning the residuals based on boosting algorithm.

4.3. Ablation Studies

Visualization result. We employed the attention key to encapsulate semantic information in image tokens, although any of these vectors can serve this purpose. Indeed, both the attention query and value contain information about the similarity between tokens in self-attention, as depicted in Figure 3. The figure illustrates the result of clustering the image tokens using the query, key, and value of attention. We examine the effectiveness of the query, key, and value through empirical investigations, finally determining that the attention key serves as the query of the prompt. The classification ability of the attention key applies not only to objects in the ground truth but also to the background and other objects. Therefore, we believe that the attention key can not only replace the class token but also enhance it.

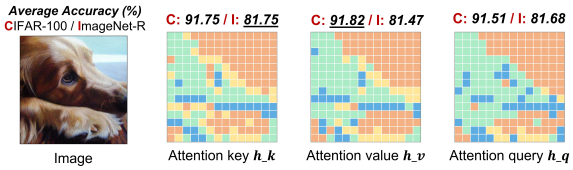


Figure 3. **Prompt query as attention features.** We present the clustering results and performance of image tokens for the query-key-value of an attention as a prompt query.

Table 9. **Effects of each component.** We report the performance of each component in our method and overall performance.

Method	CIFAR-100		ImageNet-R	
	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
Baseline	84.06	76.95	73.75	66.93
w/ image token-level prompting	89.17	83.14	78.54	72.40
w/ semantic prompt matching	91.50	87.28	77.85	72.13
w/ both (I-Prompt)	91.75	87.63	81.75	75.73

Effects of each component. We show the experimental results for each component of the proposed method in Table 9 to investigate the effect of each component on the performance. The proposed method consists of semantic prompt matching and image token-level prompts. The baseline is a structure consisting of only task-specific prompts in Dualprompt, with the addition of prefix tuning for the task-specific prompt pool. We compare the performance with image token-level prompting, which directly changes the image token, and with semantic prompt matching, which selects prompts based on internal information rather than task. Both improve performance over the baseline, with the best performance achieved when both are applied.

5. Conclusion

In this paper, we present a novel task-agnostic prompting method to address catastrophic forgetting problem in continual learning. Instead of existing task-dependent approaches that rely on task-selection, we focus on the semantic features of the images themselves. As a result, our method not only resolves the negative effects of incorrect task-selection, but also improves training efficiency by compressing the prompt selection process into a single forward pass. Our extensive empirical studies, including both task-balanced and task-imbalanced scenarios, provide in-depth insights into task-agnostic approach in prompt-based continual learning, affirming our method's effectiveness. We hope that our approach serve as a valuable groundwork for moving towards various continual learning scenarios.

Acknowledgements. This work is partially supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (RS-2023-00236245, Development of Perception/Planning AI SW for Seamless Autonomous Driving in Adverse Weather/Unstructured Environment) and NRF-2022R1A2C1091402.

References

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 139–154, 2018. 2

[2] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. *International Conference on Learning Representations*, 2023. 2, 3

[3] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 532–547, 2018. 2, 8

[4] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019. 1

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020. 2

[6] Mohsen Fayyaz, Soroush Abbasi Koohpayegani, Farnoush Rezaei Jafari, Sunando Sengupta, Hamid Reza Vaezi Joze, Eric Sommerlade, Hamed Pirsiavash, and Jürgen Gall. Adaptive token sampling for efficient vision transformers. In *In Proceedings of the European Conference on Computer Vision*, pages 396–414. Springer, 2022. 3

[7] Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*, 2013. 1

[8] Dipam Goswami, Yuyang Liu, Bartłomiej Twardowski, and Joost van de Weijer. Fecam: Exploiting the heterogeneity of class distributions in exemplar-free continual learning. *Advances in Neural Information Processing Systems*, 36, 2024. 7

[9] Saurabh Goyal, Anamitra Roy Choudhury, Saurabh Raje, Venkatesan Chakaravarthy, Yogish Sabharwal, and Ashish Verma. Power-bert: Accelerating bert inference via progressive word-vector elimination. In *International Conference on Machine Learning*, pages 3690–3699. PMLR, 2020. 3

[10] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349, 2021. 2, 5

[11] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021. 2, 5

[12] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *In Proceedings of the European Conference on Computer Vision*, pages 709–727. Springer, 2022. 3

[13] Muhammad Gul Zain Ali Khan, Muhammad Ferjad Naeem, Luc Van Gool, Didier Stricker, Federico Tombari, and Muhammad Zeshan Afzal. Introducing language guidance in prompt-based continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11463–11473, 2023. 3

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[15] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 2

[16] Hyunseo Koh, Dahyun Kim, Jung-Woo Ha, and Jonghyun Choi. Online continual learning on class incremental blurry task configuration with anytime inference. In *International Conference on Learning Representations*, 2022. 7

[17] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009. 2, 5

[18] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. 3, 5

[19] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online, Aug. 2021. Association for Computational Linguistics. 3, 5

[20] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental learning without forgetting. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 12245–12254, 2020. 2

[21] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017. 5

[22] Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[23] Dmitrii Marin, Jen-Hao Rick Chang, Anurag Ranjan, Anish Prabhu, Mohammad Rastegari, and Oncel Tuzel. Token pooling in vision transformers. *arXiv preprint arXiv:2110.03860*, 2021. 2, 3

[24] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. 1

[25] Mark D McDonnell, Dong Gong, Amin Parvaneh, Ehsan Abbasnejad, and Anton van den Hengel. Ranpac: Random projections and pre-trained models for continual learning. *Advances in Neural Information Processing Systems*, 36, 2024. 3, 7

[26] Lingchen Meng, Hengduo Li, Bor-Chun Chen, Shiyi Lan, Zuxuan Wu, Yu-Gang Jiang, and Ser-Nam Lim. Adavit: Adaptive vision transformers for efficient image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12309–12318, 2022. 3, 7

[27] Jun-Yeong Moon, Keon-Hee Park, Jung Uk Kim, and Gyeong-Moon Park. Online class incremental learning on stochastic blurry task boundary via mask and visual prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11731–11741, 2023. 7

[28] Grégoire Petit, Adrian Popescu, Hugo Schindler, David Picard, and Bertrand Delezoide. Fetril: Feature translation for exemplar-free class-incremental learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3911–3920, 2023. 1

[29] Ameya Prabhu, Shiven Sinha, Ponnurangam Kumaraguru, Philip HS Torr, Ozan Sener, and Puneet K Dokania. Randumb: A simple approach that questions the efficacy of continual representation learning. *arXiv preprint arXiv:2402.08823*, 2024. 3

[30] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 524–540. Springer, 2020. 2

[31] Anastasia Razdaibiedina, Yuning Mao, Rui Hou, Madian Khabsa, Mike Lewis, and Amjad Almahairi. Progressive prompts: Continual learning for language models. In *International Conference on Learning Representations*, 2023. 4

[32] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 1, 2, 5

[33] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 5

[34] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 2

[35] James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11909–11919, 2023. 1, 3, 4, 5, 6

[36] Kihyuk Sohn, Huiwen Chang, José Lezama, Luisa Polania, Han Zhang, Yuan Hao, Irfan Essa, and Lu Jiang. Visual prompt tuning for generative transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19840–19851, June 2023. 3

[37] Hai-Long Sun, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Pilot: A pre-trained model-based continual learning toolbox. *arXiv preprint arXiv:2309.07117*, 2023. 5

[38] Cheng-Hao Tu, Zheda Mai, and Wei-Lun Chao. Visual query tuning: Towards effective usage of intermediate representations for parameter and memory efficient transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7725–7735, June 2023. 3

[39] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. *The Caltech-UCSD Birds-200-2011 Dataset*. 7 2011. 2, 5

[40] Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Foster: Feature boosting and compression for class-incremental learning. In *In Proceedings of the European conference on computer vision*, pages 398–414. Springer, 2022. 2, 5

[41] Yabin Wang, Zhiwu Huang, and Xiaopeng Hong. S-prompts learning with pre-trained transformers: An occam's razor for domain incremental learning. *Advances in Neural Information Processing Systems*, 35:5682–5695, 2022. 1, 2, 3, 4, 5, 7

[42] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *In Proceedings of the European Conference on Computer Vision*, pages 631–648. Springer, 2022. 1, 2, 3, 5, 6, 7

[43] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022. 1, 3, 5, 6

[44] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 374–382, 2019. 1, 2, 5

[45] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pages 3987–3995. PMLR, 2017. 2

[46] Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. Slca: Slow learner with classifier alignment for continual learning on a pre-trained model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 19148–19158, October 2023. 7

[47] Da-Wei Zhou, Hai-Long Sun, Han-Jia Ye, and De-Chuan Zhan. Expandable subspace ensemble for pre-trained model-based class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23554–23564, 2024. 3

[48] Da-Wei Zhou, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. *arXiv preprint arXiv:2303.07338*, 2023. 3, 7

[49] Da-Wei Zhou, Yuanhan Zhang, Jingyi Ning, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Learning without forgetting for vision-language models. *arXiv preprint arXiv:2305.19270*, 2023. 3

[50] Fei Zhu, Zhen Cheng, Xu yao Zhang, and Cheng lin Liu. Class-incremental learning via dual augmentation. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. 1

[51] Fei Zhu, Xu-Yao Zhang, Chuang Wang, Fei Yin, and Cheng-Lin Liu. Prototype augmentation and self-supervision for incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5871–5880, 2021. 1

[52] Kai Zhu, Wei Zhai, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. Self-sustaining representation expansion for non-exemplar class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9296–9305, 2022. 1